

BBBBBBBBBBBB		AAAAAAA		SSSSSSSSSS		RRRRRRRRRR		TTTTTTTTTT		LLL
BBBBBBBBBBBB		AAAAAAA		SSSSSSSSSS		RRRRRRRRRR		TTTTTTTTTT		LLL
BBBBBBBBBBBB		AAAAAAA		SSSSSSSSSS		RRRRRRRRRR		TTTTTTTTTT		LLL
BBB	BBB	AAA	AAA	SSS		RRR	RRR	TTT		LLL
BBB	BBB	AAA	AAA	SSS		RRR	RRR	TTT		LLL
BBB	BBB	AAA	AAA	SSS		RRR	RRR	TTT		LLL
BBB	BBB	AAA	AAA	SSS		RRR	RRR	TTT		LLL
BBB	BBB	AAA	AAA	SSS		RRR	RRR	TTT		LLL
BBB	BBB	AAA	AAA	SSS		RRR	RRR	TTT		LLL
BBBBBBBBBBBB		AAA	AAA	SSSSSSSS		RRRRRRRRRR		TTT		LLL
BBBBBBBBBBBB		AAA	AAA	SSSSSSSS		RRRRRRRRRR		TTT		LLL
BBBBBBBBBBBB		AAA	AAA	SSSSSSSS		RRRRRRRRRR		TTT		LLL
BBB	BBB	AAAAAAAAAAAA			SSS	RRR	RRR	TTT		LLL
BBB	BBB	AAAAAAAAAAAA			SSS	RRR	RRR	TTT		LLL
BBB	BBB	AAAAAAAAAAAA			SSS	RRR	RRR	TTT		LLL
BBB	BBB	AAA	AAA		SSS	RRR	RRR	TTT		LLL
BBB	BBB	AAA	AAA		SSS	RRR	RRR	TTT		LLL
BBB	BBB	AAA	AAA		SSS	RRR	RRR	TTT		LLL
BBB	BBB	AAA	AAA		SSS	RRR	RRR	TTT		LLL
BBBBBBBBBBBB		AAA	AAA	SSSSSSSS		RRR	RRR	TTT		LLLLLLLLLLLL
BBBBBBBBBBBB		AAA	AAA	SSSSSSSS		RRR	RRR	TTT		LLLLLLLLLLLL
BBBBBBBBBBBB		AAA	AAA	SSSSSSSS		RRR	RRR	TTT		LLLLLLLLLLLL


```
BBBBBBBBB      AAAAAA      SSSSSSSS      CCCCCCCC      HH      HH      AAAAAA      IIIIII      NN      NN
BBBBBBBBB      AAAAAA      SSSSSSSS      CCCCCCCC      HH      HH      AAAAAA      IIIIII      NN      NN
BB      BB      AA      AA      SS      CC      HH      HH      AA      AA      II      NN      NN
BB      BB      AA      AA      SS      CC      HH      HH      AA      AA      II      NN      NN
BB      BB      AA      AA      SS      CC      HH      HH      AA      AA      II      NNNN      NN
BBBBBBBBB      AA      AA      SSSSSS      CC      HH      HH      AA      AA      II      NNNN      NN
BBBBBBBBB      AA      AA      SSSSSS      CC      HH      HH      AA      AA      II      NN      NN
BB      BB      AAAAAAAAAA      SS      CC      HH      HH      AAAAAAAAAA      II      NN      NN
BB      BB      AAAAAAAAAA      SS      CC      HH      HH      AAAAAAAAAA      II      NN      NN
BB      BB      AA      AA      SS      CC      HH      HH      AA      AA      II      NN      NN
BB      BB      AA      AA      SS      CC      HH      HH      AA      AA      II      NN      NN
BBBBBBBBB      AA      AA      SSSSSSSS      CCCCCCCC      HH      HH      AA      AA      IIIIII      NN      NN
BBBBBBBBB      AA      AA      SSSSSSSS      CCCCCCCC      HH      HH      AA      AA      IIIIII      NN      NN

LL      IIIIII      SSSSSSSS
LL      IIIIII      SSSSSSSS
LL      II      SS
LL      II      SS
LL      II      SS
LL      II      SS
LL      II      SSSSSS
LL      II      SSSSSS
LL      II      SS
LL      II      SS
LL      II      SS
LL      II      SS
LLLLLLLLLLL      IIIIII      SSSSSSSS
LLLLLLLLLLL      IIIIII      SSSSSSSS
```



```
1 0001 0 MODULE BAS$CHAIN (
2 0002 0 IDENT = '1-005'
3 0003 0 ) =
4 0004 1 BEGIN
5 0005 1
6 0006 1 *****
7 0007 1 *
8 0008 1 * COPYRIGHT (c) 1978, 1980, 1982, 1984 BY
9 0009 1 * DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.
10 0010 1 * ALL RIGHTS RESERVED.
11 0011 1 *
12 0012 1 * THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED
13 0013 1 * ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE
14 0014 1 * INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER
15 0015 1 * COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
16 0016 1 * OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY
17 0017 1 * TRANSFERRED.
18 0018 1 *
19 0019 1 * THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE
20 0020 1 * AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT
21 0021 1 * CORPORATION.
22 0022 1 *
23 0023 1 * DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS
24 0024 1 * SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
25 0025 1 *
26 0026 1 *
27 0027 1 *****
28 0028 1
29 0029 1
30 0030 1 ++
31 0031 1 FACILITY: VAX-11 BASIC Miscellaneous
32 0032 1
33 0033 1 ABSTRACT:
34 0034 1
35 0035 1 This module causes the current program to end and another
36 0036 1 program to be run in its place.
37 0037 1
38 0038 1 ENVIRONMENT: VAX-11 User Mode
39 0039 1
40 0040 1 AUTHOR: John Sauter, CREATION DATE: 13-NOV-1979
41 0041 1
42 0042 1 MODIFIED BY:
43 0043 1
44 0044 1 1-001 - Original. JBS 15-NOV-1979
45 0045 1 1-002 - Make the entry point global. JBS 28-NOV-1979
46 0046 1 1-003 - CHAIN to a non-existent file should generate an error,
47 0047 1 not just return to DCL. PLL 17-Nov-1981
48 0048 1 1-004 - If LIB$RUN_PROGRAM returns an error, signal this error
49 0049 1 to the user instead of the Basic message 'not implemented.'
50 0050 1 PLL 16-Jun-1982
51 0051 1 1-005 - signal ILLFILNAM when $PARSE fails, not FATSYSIO. MDL 27-Apr-1983
52 0052 1 --
53 0053 1
54 0054 1 !<BLF/PAGE>
```



```
56 0055 1 |
57 0056 1 | SWITCHES:
58 0057 1 |
59 0058 1 |
60 0059 1 SWITCHES ADDRESSING_MODE (EXTERNAL = GENERAL, NONEXTERNAL = WORD_RELATIVE);
61 0060 1 |
62 0061 1 |
63 0062 1 | LINKAGES:
64 0063 1 |
65 0064 1 |     NONE
66 0065 1 |
67 0066 1 | TABLE OF CONTENTS:
68 0067 1 |
69 0068 1 |
70 0069 1 FORWARD ROUTINE
71 0070 1     BASSCHAIN : NOVALUE;
72 0071 1
73 0072 1 |
74 0073 1 | INCLUDE FILES:
75 0074 1 |
76 0075 1 |
77 0076 1 REQUIRE 'RTLIN:RTLPSECT';
78 0171 1 LIBRARY 'RTLSTARLE';
79 0172 1
80 0173 1 | MACROS:
81 0174 1 |
82 0175 1 |     NONE
83 0176 1 |
84 0177 1 | EQUATED SYMBOLS:
85 0178 1 |
86 0179 1 |     NONE
87 0180 1 |
88 0181 1 | PSECTS:
89 0182 1 |
90 0183 1 DECLARE_PSECTS (BAS);
91 0184 1
92 0185 1 | OWN STORAGE:
93 0186 1 |
94 0187 1 |     NONE
95 0188 1 |
96 0189 1 | EXTERNAL REFERENCES:
97 0190 1 |
98 0191 1 |
99 0192 1 EXTERNAL ROUTINE
100 0193 1     BASS$SIGNAL : NOVALUE,
101 0194 1     BASS$STOP : NOVALUE,
102 0195 1     LIB$STOP : NOVALUE,
103 0196 1     LIB$RUN_PROGRAM;
104 0197 1
105 0198 1 |
106 0199 1 | +
107 0200 1 | The following are the error codes used in this module.
108 0201 1 | -
109 0202 1 EXTERNAL LITERAL
110 0203 1     BASS$K_ILLFILNAM : UNSIGNED (8),
111 0204 1     BASS$K_FATSYSIO : UNSIGNED (8),
112 0205 1     BASS$K_CANFINFIL : UNSIGNED (8),
```

! End this program and start another

! Macros for defining PSECTS

! Declare psepts for BASS\$ facility

! Signal an error
! Signal a fatal BASIC error
! Signal a fatal error
! Run a program! Illegal File Name
! Fatal System I/O error
! Can't find file or account

BAS\$CHAIN
1-005

K 2
16-Sep-1984 00:05:06
14-Sep-1984 11:54:45

VAX-11 Bliss-32 V4.0-742
[BASRTL.SRC]BASCHAIN.B32;1

Page 3
(2)

: 113
: 114

0206 1
0207 1

BAS\$K_NOTIMP : UNSIGNED (8);

! Not implemented


```
116 0208 1 GLOBAL ROUTINE BASSCHAIN (
117 0209 1     PGM_NAME,
118 0210 1     LINE_NO
119 0211 1 ) : NOVALUE =
120 0212 1
121 0213 1 !++
122 0214 1 FUNCTIONAL DESCRIPTION:
123 0215 1
124 0216 1     Stop running this program and start running another.
125 0217 1
126 0218 1 CALLING SEQUENCE:
127 0219 1
128 0220 1     CALL BASSCHAIN (PGM_NAME.rt.dx, LINE_NO.rl.r)
129 0221 1
130 0222 1 FORMAL PARAMETERS:
131 0223 1
132 0224 1     PGM_NAME.rt.dx The name of the program to be run in place of this one.
133 0225 1     LINE_NO.rl.r   The line number to start the program at
134 0226 1
135 0227 1 IMPLICIT INPUTS:
136 0228 1
137 0229 1     NONE
138 0230 1
139 0231 1 IMPLICIT OUTPUTS:
140 0232 1
141 0233 1     NONE
142 0234 1
143 0235 1 ROUTINE VALUE:
144 0236 1 COMPLETION CODES:
145 0237 1
146 0238 1     NONE
147 0239 1
148 0240 1 SIDE EFFECTS:
149 0241 1
150 0242 1     Either signals or never returns to its caller.
151 0243 1
152 0244 1 !--
153 0245 1
154 0246 2 BEGIN
155 0247 2
156 0248 2 LOCAL
157 0249 2     FAB_BLOCK : $FAB_DECL,
158 0250 2     FAB : REF $FAB_DECL,
159 0251 2     NAM_BLOCK : $NAM_DECL,
160 0252 2     EXP_NAME : BLOCK[NAM$C_MAXRSS, BYTE],
161 0253 2     STATUS;
162 0254 2
163 0255 2 BIND
164 0256 2     DEFAULT_NAME = UPLIT ('*.EXE');
165 0257 2
166 0258 2 MAP
167 0259 2     PGM_NAME : REF BLOCK [0, BYTE];
168 0260 2
169 0261 2 BUILTIN
170 0262 2     NULLPARAMETER;
171 0263 2
172 0264 2 !+
```



```
173 0265 2 ! The line number feature is not implemented, so give an error
174 0266 2 ! if it is specified.
175 0267 2 !-
176 0268 2
177 0269 2 IF ( NOT NULLPARAMETER (2)) THEN BASS$$STOP (BASS$K_NOTIMP);
178 0270 2
179 0271 2 !+
180 0272 2 ! If the user tries to CHAIN to a non-existent program, this should be
181 0273 2 ! returned as ERR=5, 'can't find file ...'. Since LIB$RUN_PROGRAM will
182 0274 2 ! return to DCL level if an error other than invalid argument occurs,
183 0275 2 ! try to determine if the file exists before LIB$RUN_PROGRAM is called.
184 0276 2 !-
185 0277 2
186 0278 2 FAB = FAB_BLOCK;
187 0279 2 $FAB_INIT (FAB = .FAB);
188 0280 2 FAB [FAB$L_FNA] = .PGM_NAME [DSC$A_POINTER];
189 0281 2 FAB [FAB$B_FNS] = .PGM_NAME [DSC$W_LENGTH];
190 0282 2 FAB [FAB$L_DNA] = DEFAULT_NAME;
191 0283 2 FAB [FAB$B_DNS] = 5;
192 0284 2
193 0285 2 $NAM_INIT (NAM = NAM_BLOCK);
194 0286 2 NAM_BLOCK [NAM$L_RSA] = NAM_BLOCK [NAM$L_ESA] = EXP NAME;
195 0287 2 NAM_BLOCK [NAM$B_RSS] = NAM_BLOCK [NAM$B_ESS] = NAM$C_MAXRSS;
196 0288 2
197 0289 2 FAB [FAB$L_NAM] = NAM_BLOCK;
198 0290 2
199 0291 2 !+
200 0292 2 ! Do not use $OPEN to look for the file because if the protection code
201 0293 2 ! is execute only, a privilege violation will occur.
202 0294 2 !-
203 0295 2
204 0296 2 STATUS = $PARSE (FAB = .FAB);
205 0297 2 IF (NOT .STATUS) THEN BASS$$STOP (BASS$K_ILLFILNAM);
206 0298 2
207 0299 2 STATUS = $SEARCH (FAB = .FAB);
208 0300 2 SELECTONE .STATUS OF
209 0301 2 SET
210 0302 2 [RMS$_NORMAL]:
211 0303 2 ! file exists
212 0304 2 [RMS$_FNF]:
213 0305 2 BASS$$SIGNAL (BASS$K_CANFINFIL); ! file not found
214 0306 2 [OTHERWISE]:
215 0307 2 BASS$$STOP (BASS$K_FATSYSIO_); ! random RMS error
216 0308 2 TES;
217 0309 2
218 0310 2
219 0311 2 STATUS = LIB$RUN_PROGRAM (.PGM_NAME);
220 0312 2 !+
221 0313 2 ! LIB$RUN_PROGRAM should never return. If it does, we have an
222 0314 2 ! error of some kind.
223 0315 2 !-
224 0316 2 LIB$STOP (.STATUS);
225 0317 2
226 0318 1 END; ! end of BASS$CHAIN
```

.TITLE BASS\$CHAIN

.IDENT \1-005\

.PSECT _BASSCODE, NOWRT, SHR, PIC, 2

00 00 00 45 58 45 2E 2A 00000 P.AAA: .ASCII *.EXE\<0><0><0>

DEFAULT_NAME= P.AAA

.EXTRN BASS\$SIGNAL, BASS\$STOP
.EXTRN LIB\$STOP, LIB\$RUN_PROGRAM
.EXTRN BASSK_ILLFILNAM
.EXTRN BASSK_FATSYSIO
.EXTRN BASSK_CANFINFIL
.EXTRN BASSK_NOTIMP, SYSSPARSE
.EXTRN SYSSSEARCH

.ENTRY BASSCHAIN, Save R2,R3,R4,R5,R6,R7,R8

MOVAB BASS\$STOP, R8

MOVAB -432(SP), SP

CMPB (AP), #2

BLSSU 1\$

TSTL 8(AP)

BEQL 1\$

MOVZBL #BASSK_NOTIMP, -(SP)

CALLS #1, BASS\$STOP

MOVAB FAB_BLOCK, FAB

MOVCS #0, (SP), #0, #80, (FAB)

MOVW #20483, (FAB)

MOVB #2, 22(FAB)

MOVB #2, 31(FAB)

MOVL PGM_NAME, R7

MOVL 4(R7), 44(FAB)

MOVB (R7), 52(FAB)

MOVAB DEFAULT_NAME, 48(FAB)

MOVB #5, 53(FAB)

MOVCS #0, (SP), #0, #96, \$RMS_PTR

MOVW #24578, \$RMS_PTR

MOVAB EXP_NAME, R0

MOVL R0, NAM_BLOCK+12

MOVL R0, NAM_BLOCK+4

MNEGB #1, NAM_BLOCK+10

MNEGB #1, NAM_BLOCK+2

MOVAB NAM_BLOCK, 40(FAB)

PUSHL FAB

CALLS #1, SYSSPARSE

MOVL R0, STATUS

BLBS STATUS, 2\$

MOVZBL #BASSK_ILLFILNAM, -(SP)

CALLS #1, BASS\$STOP

PUSHL FAB

CALLS #1, SYSSSEARCH

MOVL R0, STATUS

CMPL STATUS, #65537

BEQL 4\$

CMPL STATUS, #98962

BNEQ 3\$

0050	8F	00	58	00000000G	00	01FC	00000		0208	
			5E	FE50	CE	9E	00002			
			02		6C	91	0000E			
					0C	1F	00011			
				08	AC	D5	00013			
					07	13	00016			
			7E	00G	8F	9A	00018			
			68		01	FB	0001C			
			56	B0	AD	9E	0001F	1\$:	0278	
			6E		00	2C	00023		0279	
					66		0002A			
			66	5003	8F	B0	0002B			
		16	A6		02	90	00030			
		1F	A6		02	90	00034			
			57	04	AC	D0	00038		0280	
		2C	A6	04	A7	D0	0003C			
		34	A6		67	90	00041		0281	
		30	A6	B0	AF	9E	00045		0282	
		35	A6		05	90	0004A		0283	
0060	8F	00	6E		00	2C	0004E		0285	
				FF50	CD		00055			
			FF50	CD	6002	8F	B0	00058		
				50		6E	9E	0005F		0286
			FF5C	CD		50	D0	00062		
			FF54	CD		50	D0	00067		
			FF5A	CD		01	8E	0006C		0287
			FF52	CD		01	8E	00071		
			28	A6	FF50	CD	9E	00076		0289
						56	DD	0007C		0296
			00000000G	00	01	FB	0007E			
				52	50	D0	00085			
				07	52	E8	00088		0297	
				7E	00G	8F	9A	0008B		
				68		01	FB	0008F		
						56	DD	00092	2\$:	0299
			00000000G	00	01	FB	00094			
				52	50	D0	0009B			
			00010001	8F	52	D1	0009E		0302	
					1D	13	000A5			
			00018292	8F	52	D1	000A7		0304	
					0D	12	000AE			

Page 7
(3)

0305
0307
0311
0316
0318

```
! end of module BASSCHAIN
```

PSECT SUMMARY	
Name	Bytes Attributes
_BASSCODE	226 NOVEC,NOWRT, RD, EXE, SHR, LCL, REL, CON, PIC,ALIGN(2)

Library Statistics					
File	----- Total	Symbols Loaded	----- Percent	Pages Mapped	Processing Time
\$255\$DUA28:[SYSLIB]STARLET.L32;1	9776	61	0	581	00:01.0

```

;                                COMMAND QUALIFIERS
;    BLISS/CHECK=(FIELD,INITIAL,OPTIMIZE)/NOTRACE/LIS=LIS$:BASCHAIN/OBJ=OBJ$:BASCHAIN MSRC$:BASCHAIN/UPDATE=(ENH$:BASCHAIN)

```

```

: Size:                218 code + 8 data bytes
: Run Time:            00:07.3
: Elapsed Time:        00:17.2
: Lines/CPU Min:       2639
: Lexemes/CPU-Min:    27950
: Memory Used:         101 pages

```


BASSCHAIN
1-005

C 3
16-Sep-1984 00:05:06

VAX-11 Bliss-32 V4.0-742

Page 8

; Compilation Complete

0020 AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY

